Designing Gestures for Continuous Sonic Interaction

Atau Tanaka, Balandino Di Donato, and
Michael Zbyszyński
Embodied Audiovisual Interaction Group
Goldsmiths, University of London
SE14 6NW, London, UK
[a.tanaka, b.didonato,
m.zbyszynski]@gold.ac.uk

Geert Roks
Music and Technology Department
HKU University of the Arts
3582 VB, Utrecht, Netherlands
geertrocks@gmail.com

ABSTRACT

We present a system that allows users to try different ways to train neural networks and temporal modelling to associate gestures with time-varying sound. We created a software framework for this and evaluated it in a workshop-based study. We build upon research in sound tracing and mapping-by-demonstration to ask participants to design gestures for performing time-varying sounds using a multimodal, inertial measurement (IMU) and muscle sensing (EMG) device. We presented the user with two classical techniques from the literature, Static Position regression and Hidden Markov based temporal modelling, and propose a new technique for capturing gesture anchor points on the fly as training data for neural network based regression, called Windowed Regression. Our results show trade-offs between accurate, predictable reproduction of source sounds and exploration of the gesture-sound space. Several users were attracted to our windowed regression technique. This paper will be of interest to musicians engaged in going from sound design to gesture design and offers a workflow for interactive machine learning.

Author Keywords

Sonic Interaction Design, Interactive Machine Learning, Gestural Interaction

CCS Concepts

•Human-centered computing → Empirical studies in interaction design; •Applied computing → Sound and music computing;

1. INTRODUCTION

Designing gestures for the articulation of dynamic sound synthesis is a key part of the preparation of a performance with a DMI. Traditionally this takes place through a careful and manual process of mapping. Strategies for mapping, including "one-to-many" and "many-to-one" [15] are fundamental techniques in NIME. The field of embodied music cognition looks at the relationship between corporeal action and music [17]. The notion of sonic affordances draws upon the notion of affordance from environmental psychology [13] to look at how a sound may invite action [1].

Sound tracing is an exercise where a sound is given as a stimulus to study evoked gestural response [3]. Sound tracing has been used as a starting point for techniques of "mapping-by-demonstration" [12]. While these studies look at the articulation of gesture in response to sounds, they focus on evoked gesture. In the field of sonic interaction design, embodied interaction has been used to design sounds. This includes techniques applying interactive technologies to traditions of Foley, or by vocalisation [7] and invoke the body in the design of sounds.

The synthesis of time-varying sounds and the exploration of timbral spaces is a practice at the heart of computer music research. Wessel's seminal work in the field defines timbre space in a Cartesian plane [22]. Momeni has proposed interactive techniques for exploring timbre spaces [18].

Neural networks can be trained for regression tasks by providing examples of inputs associated with desired outputs. In systems for interactive machine learning, like Wekinator [9], this is implemented by associating positions in 3D space to synthesised sound output. Once a model is trained, the user performs by moving between (and beyond) the example positions to create dynamic sound by gestures. While performance is dynamic, the training is based on poses associated with sound synthesis parameters that are fixed for each input example. Here we call this approach "static regression."

Time-varying gestures can be modelled by probabilistic approaches, such as Hidden Markov Models. In performance, live input is compared to transition states of the model, allowing the algorithm to track where in the example gesture the input is. This approach is commonly referred to as temporal modelling.

We present a system for designing gestures to perform time-varying synthesised sound. It extends the notion of mapping-by-demonstration in a practical setting by enabling users to capture gesture while listening to sound, and then to train different machine learning models. It associates the authoring of gesture to interactive sound synthesis and in so doing, explores the connection between sound design and gesture design. The technique uses commonly available tools for musical performance and machine learning and assumes no specialist knowledge of machine learning. It will be useful for artists wishing to create gestures for interactive music performances in which gestural input articulates dynamic synthesised sound where the association of gesture and sound is not made by direct mapping, but mediated by machine learning.

We propose an automated technique for training a neural network with a windowed set of anchor points captured on the fly from a dynamic gesture made in response to a sound tracing stimulus. We call this technique *Windowed Regression* and evaluate it alongside static regression and temporal modelling to gain insight into its usefulness in a

NIME'19, June 3-6, 2019, Federal University of Rio Grande do Sul, Porto Alegre, Brazil.

gesture design task.

This paper is organised as follows. In the next section, we survey related work in the area of machine learning of musical gesture. In Section 3, we present the architecture of our system, its techniques of sound design, machine learning and the proposed workflow. Section 4 presents a workshop-based evaluation. This is followed by a discussion to gather insight from user experiences.

2. RELATED WORK

Fiebrink established an interactive machine learning (IML) workflow for musicians carrying out classification and regression tasks with gestural input driving sound synthesis output where users are able to edit, delete, and add to training datasets interactively [9]. In a typical workflow with Wekinator, a regression task would be trained by static postures. Scurto [10] proposes a method of extracting examples from dynamic performances in response to sonic stimuli.

Caramiaux [3] uses Canonical Correlation Analysis to study evoked gestures in response to sound stimuli and explores the different movement-sound relationships evoked by "causal" and "non-causal" sounds [5]. In the latter, users trace the sound's frequency/amplitude morphology.

Nymoen [19] conducted a large scale sound tracing study relating gesture features (position, velocity, acceleration) to sound features such as loudness, brightness and pitch, and found a direct relationship between spectral centroid and vertical motion. When the movement of pitch was opposite to the motion of the spectral centroid, participants were more likely to move their hands following the pitch. When listening to noisy sounds, participants performed gestures that were characterised by a higher acceleration.

Françoise [11] studied different probabilistic models in mapping-by-demonstration. He uses two kinds of modelling, Gaussian Mixture Models (GMM), and Hierarchical Hidden Markov Models (HHMM) and uses each in two different ways: 1.) to model gesture itself (single mode), and 2.) to model gesture along with the associated sound (multimodal). GMMs provide a probabilistic classification of gesture or regression based on a gesture-sound relationship, while HMM-based approaches create temporal models either of the gesture by itself or of the gesture-sound association. We adopt his HHMM approach as one of the algorithms used in our proposed system.

There are an increasing number of machine learning software packages for interactive music applications [8] [14] [2] [20] [23]. While these tools expose machine learning technologies to artists, they still require configuration and integration into a music composition or performance system. One part of our proposed system is a scriptable interface where the user can assign gesture features to feed Wekinator, and select synthesis parameters to be controlled by Wekinator's output. We provide a generic Wekinator project that runs in the background that is controlled by our system.

3. THE SYSTEM

We developed our system using Cycling'74 Max, Fiebrink's Wekinator for neural network regression, and the HHMM object from IRCAM's MuBu library for temporal modelling.

3.1 Architecture

Our system is modular, comprised of three (3) blocks:

1. A scriptable sensor input and gesture feature extraction module

2. A scriptable synthesiser controller with breakpoint envelopes to dynamically send selected parameters to the machine learning module

3. A machine learning training module to capture gesture training sets and remotely control Wekinator

3.1.1 Sensor input & feature extraction

For this study, we capture gesture using a Thalmic Labs Myo, using its electromyogram (EMG) muscle sensing and inertial measurement unit (IMU) gross movement and orientation sensing. To extract orientation from the IMU, we capture Euler Angles (x, y, z) of the forearm. We calculate the first order differences (x_d, y_d, z_d) of these angles, which are correlated with direction and speed of displacement, and augment our regression feature vector with historical data.

We detect gesture *power* [4] by tracking muscle exertion, following the amplitude envelope of four (of the Myo's 8) EMG channels with a Bayesian filter [21].

The *sendrcv* scripting system we propose allows the user to select any number of features to be sent to Wekinator as inputs. In this way, the proposed system is not specific to the Myo and can be used with other sensors and input feature extraction algorithms.

3.1.2 Synthesizer playback

We used a general purpose software synthesizer, SCP by Manuel Poletti. This synthesizer is controlled by our breakpoint envelope-based playback system. We chose to design sounds that transition between four fixed anchor points (start, two intermediate points, and end) that represent fixed synthesis parameters. The envelope interpolates between these fixed points. The temporal evolution of sound is captured as different states in the breakpoint editor whose envelopes run during playback, feeding both synthesizer and Wekinator. Any of the parameters can be assigned to breakpoint envelopes to be controlled during playback.

The sounds are customisable. For the workshop, we created two sounds with granular synthesis and one sound using a looping sample synthesizer. These sound trajectories are reproduced during the gesture design and model training phases of our workflow (section 3.2). In performance a model maps sensor data to synthesis parameters, allowing users to reproduce the designed sounds or explore sonic space around the existing sounds.

3.1.3 Wekinator communications

We developed a scripting system, *sendrcv*, in Max that allows modularity and high-level use of the system. *Sendrcv* is a configurable scaling and mapping abstraction that sets up assignable sends and receives between Wekinator, the gesture features that feed it, and the synthesis parameters it controls. On input, it allows the user to select gesture features to be recorded by Wekinator. On output, each instantiation makes a bridge between a parameter in the synthesizer and the model output.

Sendrcv is invoked with network ports as arguments, allowing multiple sensor inputs and synthesizers to be used in parallel with a corresponding number of Wekinator projects. It is instantiated with a unique name so messages can be addressed specifying the gesture feature or synthesizer parameter that it feeds or is controls. It is bidirectional, allowing the use of a synthesizer's user interface or the Wekinator sliders to author sounds. The relevant range of a synthesizer parameter can be defined in the script and is normalised to a floating point value in the range, $0.0 - 1.0$. This allows a Wekinator project to be agnostic to synthesizer specifics. Other scripting features include throttling the data rate using *speedlim*, and a ramp destination time for Max's *line* object. A typical setup script is:

```
    ; 6448weki01 sendrcv mysend;
    6448weki01 arg myarg;
    6448weki01 min 0;
    6448weki01 max 127;
    6448weki01 speedlim 10;
    6448weki01 time 10;
```

The sound and gesture design workflow are described below in the section 3.3.

3.2 Machine Learning Approaches

Four different approaches to machine learning (ML) are used in the system. We provide three different ways to train neural networks for regression, each using the same algorithm and topology, but varying in the way training data are captured. A fourth method uses HHMMs for temporal modelling, which we chose because it can track progress inside of a gesture.

3.2.1 *Static Regression*

In the first approach, after designing the sound-gesture interaction through the sound tracing exercise, users segment their gestural performance into four discrete poses, or anchor points. These points coincide with breakpoints in the synthesis parameters (section 3.1.2). Training data are recorded by pairing sensor data from static poses with fixed synthesis parameters. These data are used to train a regression model, so in performance participants can explore a continuous mapping between the defined training points. We refer to this technique as *Static Regression*.

3.2.2 *Temporal Modelling*

In the second approach, we train temporal models, specifically Hierarchical Hidden Markov Models implemented with MuBu [11]. HHMMs are used to automatically segment a gesture into 10 equal-sized states, each represented by a Gaussian Mixture Model. In performance, the output of an HHMM is used to step along the synthesis parameter timeline. Here, we refer to this technique as *Temporal Modelling*.

3.2.3 *Whole Regression*

In a third approach, we train a neural network using input and output data generated during the whole duration of the sound. We call this algorithm *Whole Regression*.

3.2.4 *Windowed Regression*

Finally, we propose our method: training a neural network with gestural data and synthesis parameters from four temporal windows centred around the four fixed anchor points in the sound. Anchor points are defined as points in time where there is a breakpoint in the functions that generate synthesis parameters over time (red circles in Figure 1). This includes the beginning and end of the sound, as well as two equally spaced intermediate points. Training data are recorded during windows that are and centred around the anchor points and have a size of 1/6 of the whole duration of given sound (grey areas in Figure 1). We call this *Windowed Regression*.

3.3 Workflow

The workflow is divided into four top level activities: *Sound design, Gesture design, Machine training* and *Performance*. While we present them here in order, they and the steps within them can be carried out iteratively and interactively.

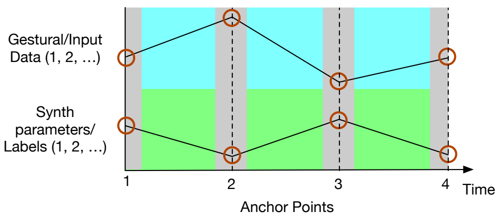
Gestural/Input Data (1, 2, ...)
Synth parameters/ Labels (1, 2, ...)
1
2
3
4
Time
Anchor Points

Figure 1: Windowed Regression. The red circles represent the four anchor points, and the grey zones show the window of recorded data around each anchor point.

3.3.1 *Sound design*

In the sound design phase of our workflow, users use their preferred synthesizer to author sounds. They select salient synthesis parameters that will be modulated in the temporal evolution of the sound. These parameters are scripted in *sendrcv.* A sound trajectory is then composed of four anchor points. The user then records these anchor points using the *Envelope* window of our system. They create a variant on their sound, select the breakpoint to which they would like to assign it $(0 - 3)$, and click *Set* (Fig. 2).

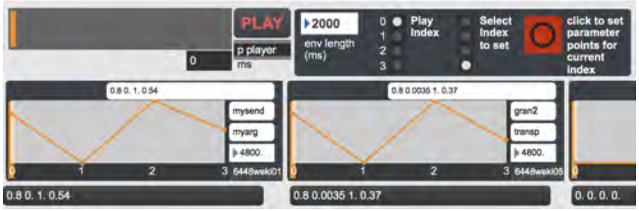

Figure 2: The Envelope window showing progress bar, sound duration, anchor point selection, set button above, and several envelopes below.

In this way, short (< 10 second) sounds can be created with dynamic parameter envelopes that are suitable for sound tracing.

3.3.2 *Gesture design*

The gesture design part of the system (Fig. 3) enables the user to choose between the different ML approaches mentioned above (Section 3.2). The user selects a sound to preview in the left part of the panel. In the current version, there are three (3) authored sounds that can be previewed, each with four (4) anchor points. The *Play* button below the progress bar changes name contextually to play the time-varying sound trajectory or one of the selected anchor points. In this way the user can conceive their gesture by sound tracing, practice executing it while listening to the sound, and find salient anchor points in the gesture that correspond to anchor points in the sound.

3.3.3 *Model training*

Once the gestures are designed, the user can train their choice of ML algorithms. Figure 4 shows the logical sequence. First, the user decides whether to work with anchor points in a static regression approach or using dynamic gesture in one of three time-based approaches. In the latter case, they choose from whole or windowed regression or temporal modelling. This part is seen in the middle pane of the interface in Fig. 3. Once the algorithm is chosen, the user proceeds with training using the right panel. The *Record* button records examples. If a dynamic algorithm is chosen, this will play the selected sound, and the user

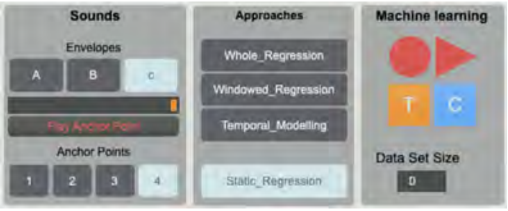

Figure 3: The machine learning training panel, with selection of sounds (with possible selection of anchor point for static regression) ***(Left)*****, Selection of ML algorithm** ***(Centre)*****, and Record, Play, Train, and Clear Dataset buttons** ***(Right)*****.**

records a training set by sound tracing, in the same way they designed the gesture. If the user has chosen Static Regression, they select an anchor point on the left, hold the pose to associated with the anchor point, and then click the *Record* button. This is repeated for each of the anchor points. At any point, the user has the possibility to Clear their recording (the **C** button) to re-take their gesture or their posture. The Data Set Size field shows the number of samples recorded. If they are happy with their recording, the user then trains a model by clicking the **T** button.

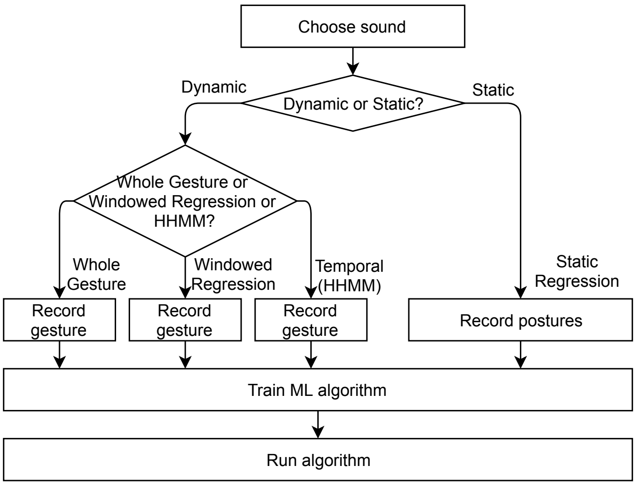

Figure 4: The machine training decision tree, where the user selects static regression, one of two types of dynamic regression, or temporal modelling.

4. EVALUATION

We organised a half-day workshop where we presented the software and asked participants to explore each approach to machine learning. We collected qualitative data in the form of video capturing participants' experience using our proposed system. Data were analysed by adopting Open and Axial Coding Methods [6].

4.1 Participants

The workshop was not meant to be a tutorial on ML techniques nor a primer on sonic interaction design. We, therefore, recruited participants who were creative practitioners in music, dance, or computational art, who had some prior exposure to topics of embodied interaction and ML. We recruited five (5) participants (3 female, 2 male). Three were Computational Arts Masters students with interest in dance technology, one was a recent undergraduate Creative Computing graduate, and one was a PhD student in live audiovisual performance.

Figure 5: A workshop participant demonstrating her gesture.

4.2 Procedure

We provided the hardware and software system on lab computers. We also provided three (3) sounds that had been prepared for the study:

A A Theremin-like whistling sound with a frequency trajectory ending in rapid vibrato

B A rhythmic sound of repeating bells where speed and pitch were modulated

C Scrubbing of a pop song where granular synthesis allowed time stretching

By providing the sounds, the workshop focused on the Gesture Design segment of the workflow described above.

We focused on Sound A, the frequency trajectory of the whistling tone. Participants listened to the sound, designing their gesture by sound tracing. They then tried Whole Regression. In the second task, participants were asked to think about breaking their gesture down into anchor points to train the system by Static Regression. Task three consisted of trying Windowed Regression and Temporal Modelling. We finished with a free exploration segment where the participants tried the other two sounds with algorithms of their choosing.

4.3 Results

Four of five participants designed a gesture for Sound A that was consistent with theory from sound tracing; they followed the amplitude/frequency morphology of the sound with sweeping arm gestures and muscle tension. One participant designed her gesture with a drawing on paper (Fig. 6). Participants tried to represent the wobbly vibrato at the end of the sound in different ways: by wiggling their fingers, flapping their hands, or making a fist. P1 commented on Whole Regression where interaction with the sound *"became embodied, it was giving me, and I was giving it."*

The participants responded differently to decomposing their gesture into anchor points for Static Regression. For P1 this meant that she *"could be more precise."* P2 identified what she called, *"natural"* points along her paper sketch as anchors. These included key points like the turn of a line, but also the middle of a smooth curve (Fig. 6). P3 felt that this technique had a less fluid response, like triggering different *"samples"*. P4 found it difficult to decompose her smooth gesture into constituent anchors: *"It was difficult to have the four anchor points... Sure the sound was divided up in different pitches but.."*. P5 felt that *"the connection between the sound and the movement was not as close [as using Whole Regression]."* P1 took this as a creative opportunity, *"I had the possibility to reinvent the transitions."*

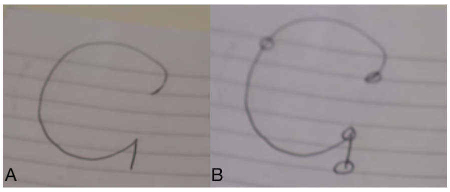

Figure 6: Gesture design by drawing. P2 in Task 1 *(Left)*, then Task 2 with anchor points *(Right)*.

With Temporal modelling, P1 seemed to track the orientation of her arm more than her hand gestures. P3 found it to be *"too discrete"* and P4 sound it *"super choppy."* P5 remarked, *"you could hear the transitions, it was less fluid than number one [Whole regression]. It was steppy."*

Three participants (P1, P3, P4) had positive reactions to our Windowed Regression technique. P1 used it with Sound B (a rhythmic bell) in a gesture consisting of waving her hand out and twisting her wrist while moving her arm from frontwards to upwards. After trying and clearing the recording four times, she perfected her gesture by paying attention to shoulder position and finger tension. P3 and P4 chose Windowed Regression with Sound C (a scrubbed and filtered sample of a pop song). P3 *"performed"* with it in a playful manner: *"What I was trying to do was... to separate out the bits."* P4 played with the *"acceleration of the gesture... because of the sound [song], that's more a continuous sound and movement, so I worked more with the acceleration."* P1 and 3 felt that this technique enabled them to reproduce the sound accurately but at the same time also to explore new sonic possibilities.

In the free exploration segment of the workshop, four out of five participants (P2, P3, P4 and P5) presented their explorations with Sound B (rhythmic bells). P5 trained a Static Regression model with different spatial positions of the arm. P3 did similarly and attempted to add striking gestures to follow the rhythmic accelerando. P2 associated speed of movement to bell triggering using Temporal Modelling. She tried with the arm in a fixed position and again by changing orientation, and felt that the latter worked better for her. P2 showed how she used the bell sound with *Whole Regression*. She performed a zig-zag like movement, and explored the quiet moments she could attain through stillness, similar to the work of Jensenius et al. [16]

Participants were interested in going beyond reproducing the sound trajectory they had traced, exploring the expressivity of a given technique and responding to variations of gesture within and outside the designed gesture. Sound B (rhythmic bell) was the most difficult sample to reproduce faithfully but gave more expressivity, P5 said *"it gave the best interaction... the most surprising results."*

5. DISCUSSION AND CONCLUSIONS

We have presented a system for designing gesture and implementing four related machine learning techniques. We presented those techniques in a workshop without giving their name or technical details on how each algorithm worked. The only indication about how static modelling differed from the dynamic techniques was that participants were asked to train the system using gesture anchor points. In this sense, this study was not a comparison of different modelling techniques. In the release version of our system[1], we expose the names of the algorithms in the UI, making a direct comparison possible.

The workflow afforded by our system enables the user, without specialist knowledge of ML and without directly configuring and operating ML algorithms, to enter into a musically productive gesture design activity following the IML paradigm. Our system is aimed at musicians and artists who might imagine incorporating embodied interaction and machine learning into a performance. The workshop participants represented such a user group: they were comfortable with digital technologies, but did not have specific technical knowledge of feature extraction or machine learning. However, they were articulate in describing what they experienced and insightful in discerning the different kinds of gesture-sound interaction each algorithm afforded.

The intuitive way in which our users explored the different algorithms means they were able to train models that did not perform as expected. Without visibility into the data and how an algorithm was processing it, it is difficult to know how to alter one's approach when training a new model. While sometimes unpredictable performance was a positive effect, it was more commonly viewed as an error. Three users (P3, P4, P5) felt that Static Regression did not result in smooth interaction. This may be due to large amounts of training data and a possible overfitting effect. We took this into consideration in a design iteration of the system. Based on this, we added an auto-stop feature in the static gesture recording so that it stops after 200 samples.

Participants on the whole confirmed findings of sound tracing studies. They followed the amplitude/frequency morphology of a sound when it was non-causal [5]. When they sought to trace a more casual type of sound such as the bell hits, they tried to make striking gestures. Such gestures would be missed by a regression algorithm. Meanwhile, a temporal model would have difficulty tracking the repetitive looping nature of such a gesture. While in the output of the neural network, modulation of the sample loop-end point caused an accelerando in the bell rhythm, a striking rhythm on input was not modelled.

Meanwhile having multiple input modalities (EMG and IMU) gave the users multiple dimensions on which to trace sound morphology. With a single modality, like motion capture in Cartesian space, it can be unclear whether a gesture like raising the arms is tracing rising frequency or amplitude or both. By using muscle tension and orientation independently, we saw that our users used the IMU to follow pitch contour, and muscle tension to follow intensity of sound – be they in amplitude or effects like the nervous vibrato at the end of the whistling Theremin-like tone. This is consistent with Nymoen's observation on the change in sound tracing strategies as users encounter noisier sounds [19]. While Nymoen sees increased acceleration, here the EMG modality allows an effort dimension in sound tracing that does not have to follow pitch or spectral centroid.

While the workshop focused on the gesture design workflow, we imagine users will be interested in designing sounds along with performance gestures, and training models accordingly. We hope our method of designing sounds with trajectories is effective. However, authoring sounds using only four anchor points may be frustrating for some. If the number of anchor points is too few, our system could be expanded to accommodate more. However, in the current version, anchor points are synchronous. It is possible that

[1] `https://gitlab.doc.gold.ac.uk/biomusic/continuous-gesture-sound-interaction`

sound designers would not want parameters to have breakpoints at the same points in time. Future development will involve integrating our system into full musical performance environments, incorporating multiple sounds and gestures, providing an interface for saving and loading models, and accounting for performance issues such as fatigue.

In demonstrations of machine learning for artists, tutorials often focus on the rapid prototyping advantages of the IML paradigm. In a desire to get artists up and running with regression and modelling techniques, examples are recorded quickly and trained on random variations of synthesizer sounds. The focus is on speed and ease of use. Scurto found that the serendipity this causes can bring a certain creative satisfaction [10]. However, we can imagine that once comfortable with the *record-train-perform-iterate* IML loop, that composers and performers will want to work with specific sounds or choreographies of movement. It is here that sound design and gesture design meet. Our system provides a sound and gesture design front end to IML that connects the two via sound tracing.

Participants in our workshop were concerned about the fluidity of response of the ML algorithms. They discussed the choice of algorithms as a trade-off between faithfully reproducing the traced sound and giving them a space of exploration to produce new, unexpected ways to articulate the sounds. In this way, they began to examine the gesture/sound affordances of the different approaches to regression and temporal modelling our system offered. We might say that this enabled them to exploit IML for a gestural exploration of Wessel's timbre space.

This paper presented a system that enabled sound and gesture design to use techniques of sound tracing and IML in authoring continuous embodied sonic interaction. It introduced established techniques of static regression and temporal modelling and proposed a hybrid approach, called Windowed Regression, to track time-varying sound and associated gesture to automatically train a neural network with salient examples. Workshop participants responded favourably to Windowed Regression, finding it fluid and expressive. They were successful in using our system in an iterative workflow to design gestures in response to dynamic, time-varying sound synthesis. We hope that this system and associated techniques will be of interest to artists preparing performances with time-based media and machine learning.

6. ACKNOWLEDGEMENT

We acknowledge our funding body H2020-EU.1.1. - EXCELLENT SCIENCE - European Research Council (ERC) - ERC-2017-Proof of Concept (PoC) - Project name: BioMusic - Project ID: 789825.

7. REFERENCES

[1] A. Altavilla, B. Caramiaux, and A. Tanaka. Towards gestural sonic affordances. In *Proc. NIME*, Daejeon, Korea, 2013.

[2] J. Bullock and A. Momeni. ml.lib: Robust, cross-platform, open-source machine learning for max and pure data. In *Proc. NIME*, pages 265–270, Baton Rouge, Louisiana, USA, 2015.

[3] B. Caramiaux, F. Bevilacqua, and N. Schnell. Towards a gesture-sound cross-modal analysis. In *Gesture in Embodied Communication and Human-Computer Interaction*, pages 158–170, Berlin, Heidelberg, 2010.

[4] B. Caramiaux, M. Donnarumma, and A. Tanaka. Understanding gesture expressivity through muscle sensing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(6):31, 2015.

[5] B. Caramiaux, P. Susini, T. Bianco, et al. Gestural embodiment of environmental sounds : an experimental study. In *Proc. NIME*, pages 144–148, Oslo, Norway, 2011.

[6] J. M. Corbin and A. L. Strauss. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory.* SAGE, Fourth edition, 2015.

[7] S. Delle Monache and D. Rocchesso. To embody or not to embody: A sound design dilemma. In *Machine Sounds, Sound Machines. XXII Colloquium of Music Informatics*, Venice, Italy, 2018.

[8] R. Fiebrink and P. R. Cook. The Wekinator: a system for real-time, interactive machine learning in music. In *Proc. ISMIR*, Utrecht, Netherlands, 2010.

[9] R. Fiebrink, P. R. Cook, and D. Trueman. Human model evaluation in interactive supervised learning. In *Proc. CHI*, pages 147–156, Vancouver, BC, Canada, 2011.

[10] R. Fiebrink and H. Scurto. Grab-and-play mapping: Creative machine learning approaches for musical inclusion and exploration. In *Proc. ICMC*, pages 12–16, 2016.

[11] J. Françoise, N. Schnell, R. Borghesi, and F. Bevilacqua. Probabilistic models for designing motion and sound relationships. In *Proc. NIME*, pages 287–292, London, UK, 2014.

[12] J. Françoise. *Motion-sound mapping by demonstration.* PhD thesis, UPMC, 2015.

[13] J. Gibson. Theory of affordances. In *The ecological approach to visual perception.* Lawrence Erlbaum Associates, 1986.

[14] N. Gillian and J. A. Paradiso. The Gesture Recognition Toolkit. *The Journal of Machine Learning Research*, 15(1):3483–3487, 2014.

[15] A. Hunt and M. M. Wanderley. Mapping performer parameters to synthesis engines. *Organised Sound*, 7(2):97–108, 2002.

[16] A. R. Jensenius, V. E. Gonzalez Sanchez, A. Zelechowska, and K. A. V. Bjerkestrand. Exploring the myo controller for sonic microinteraction. In *Proc. NIME*, pages 442–445, Copenhagen, Denmark, 2017.

[17] M. Leman. *Embodied music cognition and mediation technology.* MIT Press, 2008.

[18] A. Momeni and D. Wessel. Characterizing and controlling musical material intuitively with geometric models. In *Proc. NIME*, pages 54–62, Montreal, Canada, 2003.

[19] K. Nymoen, J. Torresen, R. I. Godøy, and A. R. Jensenius. A statistical approach to analyzing sound tracings. In *Speech, Sound and Music Processing: Embracing Research in India*, pages 120–145, Berlin, Heidelberg, 2012.

[20] A. Parkinson, M. Zbyszyński, and F. Bernardo. Demonstrating interactive machine learning tools for rapid prototyping of gestural instruments in the browser. In *Proc. Web Audio Conference*, London, UK, 2017.

[21] T. D. Sanger. Bayesian filtering of myoelectric signals. *Journal of neurophysiology*, 97(2):1839–1845, 2007.

[22] D. L. Wessel. Timbre space as a musical control structure. *Computer Music Journal*, pages 45–52, 1979.

[23] M. Zbyszyński, M. Grierson, M. Yee-King, et al. Rapid prototyping of new instruments with codecircle. In *Proc. NIME*, Copenhagen, Denmark, 2017.